ENV 790.30 - Time Series Analysis for Energy Data | Spring 2023

Assignment 6 - Due date 03/06/23

Yuxiang Ren

Directions

You should open the .rmd file corresponding to this assignment on RStudio. The file is available on our class repository on Github. And to do so you will need to fork our repository and link it to your RStudio.

Once you have the file open on your local machine the first thing you will do is rename the file such that it includes your first and last name (e.g., "LuanaLima_TSA_A06_Sp23.Rmd"). Then change "Student Name" on line 4 with your name.

Then you will start working through the assignment by **creating code and output** that answer each question. Be sure to use this assignment document. Your report should contain the answer to each question and any plots/tables you obtained (when applicable).

When you have completed the assignment, **Knit** the text and code into a single PDF file. Submit this pdf using Sakai.

R packages needed for this assignment: "xlsx" or "readxl", "ggplot2", "forecast","tseries", and "Kendall". Install these packages, if you haven't done yet. Do not forget to load them before running your script, since they are NOT default packages.

Questions

This assignment has general questions about ARIMA Models.

Packages needed for this assignment: "forecast","tseries". Do not forget to load them before running your script, since they are NOT default packages.\

```
#Load/install required package here
library(forecast); library(tseries); library(sarima)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Loading required package: stats4
```

```
##
## Attaching package: 'sarima'
```

```
## The following object is masked from 'package:stats':
##
##     spectrum
```

Q1

Describe the important characteristics of the sample autocorrelation function (ACF) plot and the partial sample autocorrelation function (PACF) plot for the following models:

- AR(2)

  Answer: The ACF of AR (2) decay exponentially with time. The PACF of AR (2) has two significant value lags, Lag 1 and Lag 2. After Lag 2, the following lags' value is small and insignificant.

- MA(1)

  Answer: The ACF of MA (1) has a significant correlation at Lag 1 only, followed by lower and non-significant lag values. The PACF shows that the value of lags will decrease exponentially.

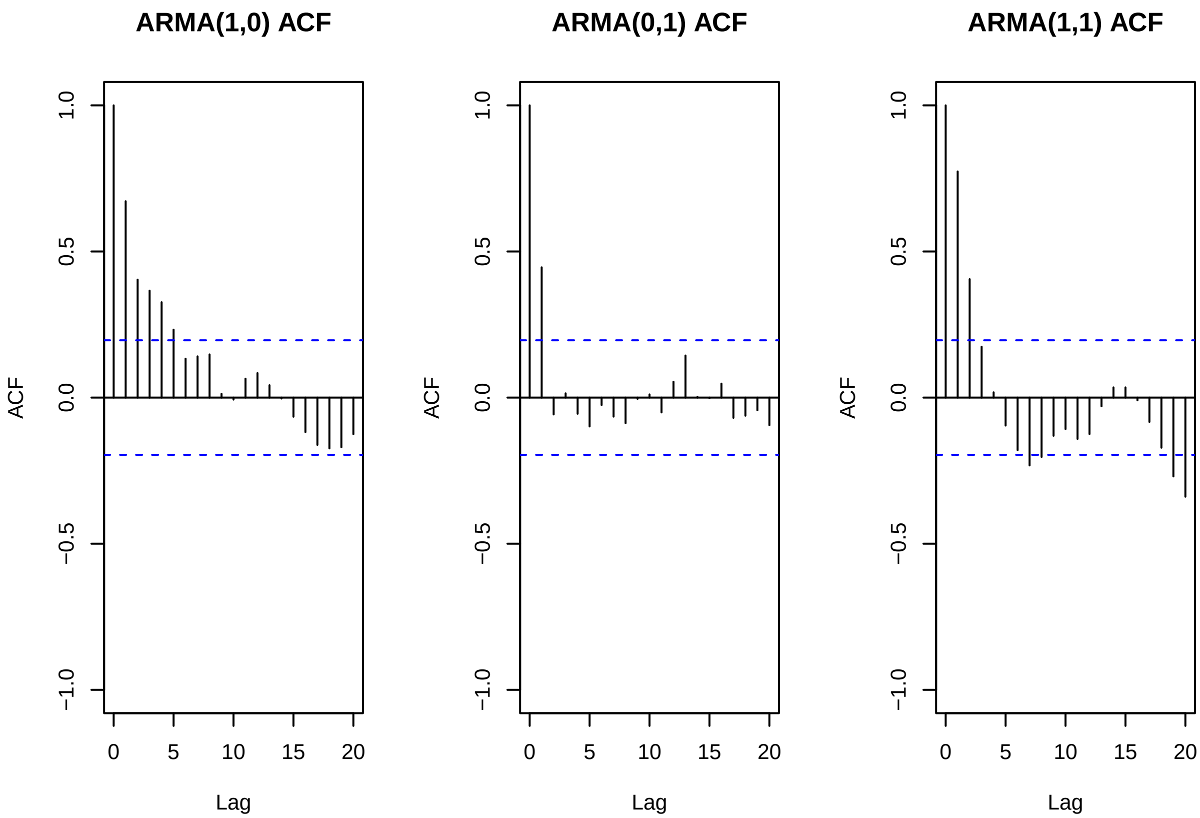
Q2

Recall that the non-seasonal ARIMA is described by three parameters ARIMA(p, d, q) where p is the order of the autoregressive component, d is the number of times the series need to be differenced to obtain stationarity and q is the order of the moving average component. If we don't need to difference the series, we don't need to specify the "I" part and we can use the short version, i.e., the ARMA(p, q). Consider three models: ARMA(1,0), ARMA(0,1) and ARMA(1,1) with parameters $\phi = 0.6$ and $\theta = 0.9$. The ϕ refers to the AR coefficient and the θ refers to the MA coefficient. Use R to generate $n = 100$ observations from each of these three models

```r
#ARMA(1,0)
set.seed(111)
ARMAmodel_1<- arima.sim(model=list(ar=0.6), n=100) #the AR coefficient is 0.6
#ARMA(0,1)
set.seed(222)
ARMAmodel_2<- arima.sim(model=list(ma=0.9), n=100) #the MA coefficient is 0.9
#ARMA(1,1)
set.seed(333)
ARMAmodel_3<- arima.sim(model=list(ar=0.6,ma=0.9), n=100)
```

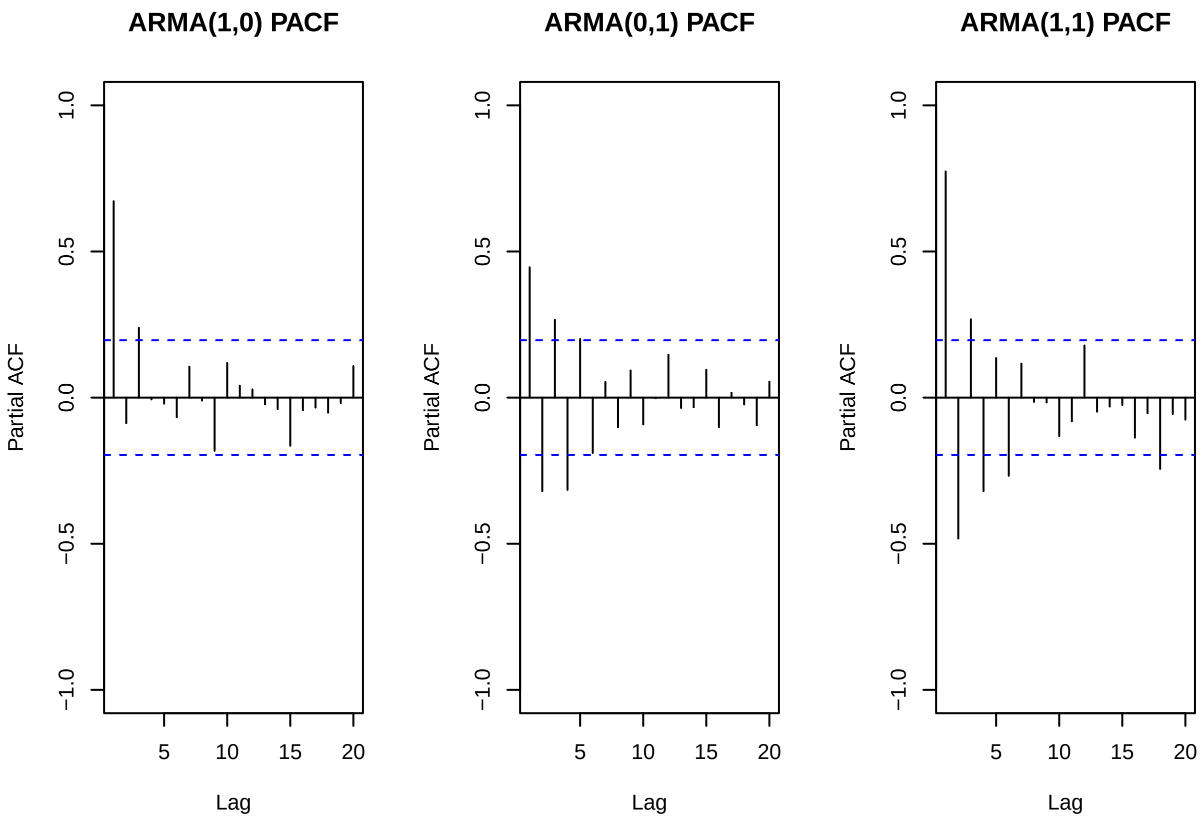
(a) Plot the sample ACF for each of these models in one window to facilitate comparison (Hint: use command $par(mfrow = c(1, 3))$ that divides the plotting window in three columns).

```r
par(mfrow = c(1,3))
acf(ARMAmodel_1, main = "ARMA(1,0) ACF", ylim=c(-1,1))
acf(ARMAmodel_2, main = "ARMA(0,1) ACF", ylim=c(-1,1))
acf(ARMAmodel_3, main = "ARMA(1,1) ACF", ylim=c(-1,1))
```



(b) Plot the sample PACF for each of these models in one window to facilitate comparison.

```
par(mfrow = c(1,3))
pacf(ARMAmodel_1, main = "ARMA(1,0) PACF", ylim=c(-1,1))
pacf(ARMAmodel_2, main = "ARMA(0,1) PACF", ylim=c(-1,1))
pacf(ARMAmodel_3, main = "ARMA(1,1) PACF", ylim=c(-1,1))
```

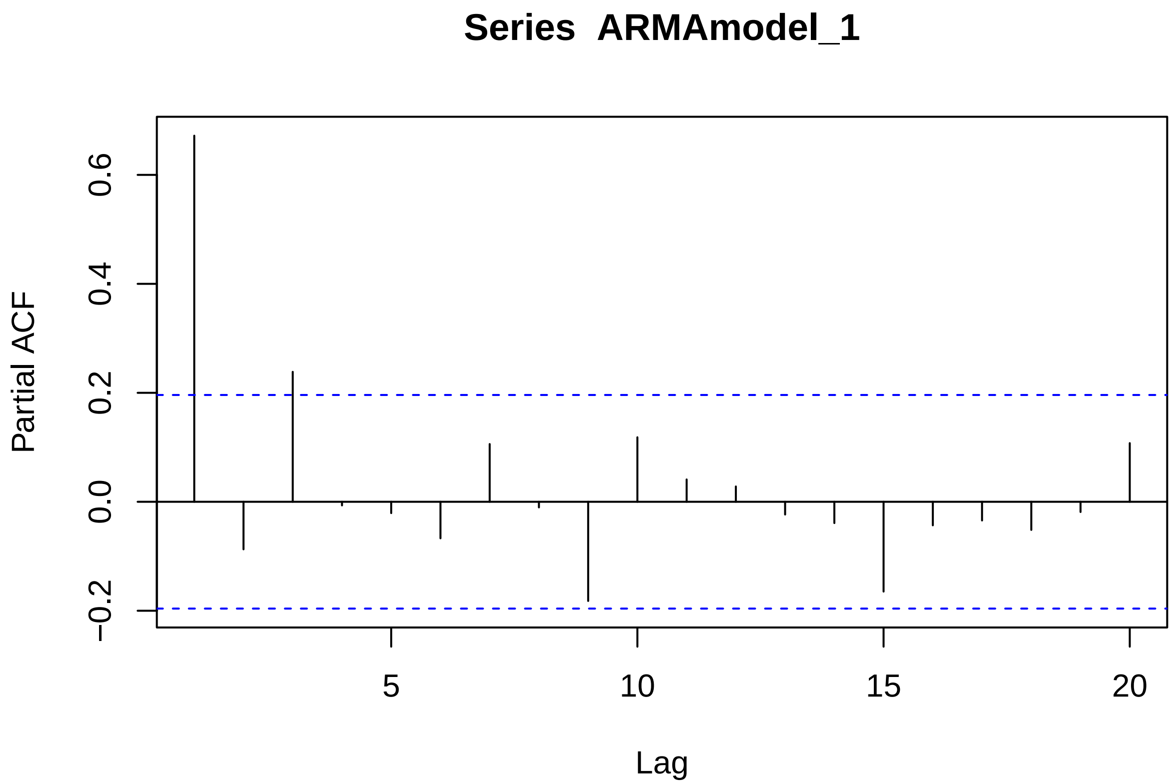


(c) Look at the ACFs and PACFs. Imagine you had these plots for a data set and you were asked to identify the model, i.e., is it AR, MA or ARMA and the order of each component. Would you be identify them correctly? Explain your answer.

Answer: If only identify the model, these plots are enough and can get a correct result. Because the identify model process only needs to judge whether the change in the lag value satisfies a specific trend. However, when identifying the order, we check the lag value, and these lag values near the significant line might affect the correctness of the judgment. For example, in AR's PACF, Lag 2 have significant value.

(d) Compare the ACF and PACF values R computed with the theoretical values you provided for the coefficients. Do they match? Explain your answer.

```
#ARMA(1,0)
ARMA10_pacf <- pacf(ARMAmodel_1)
```



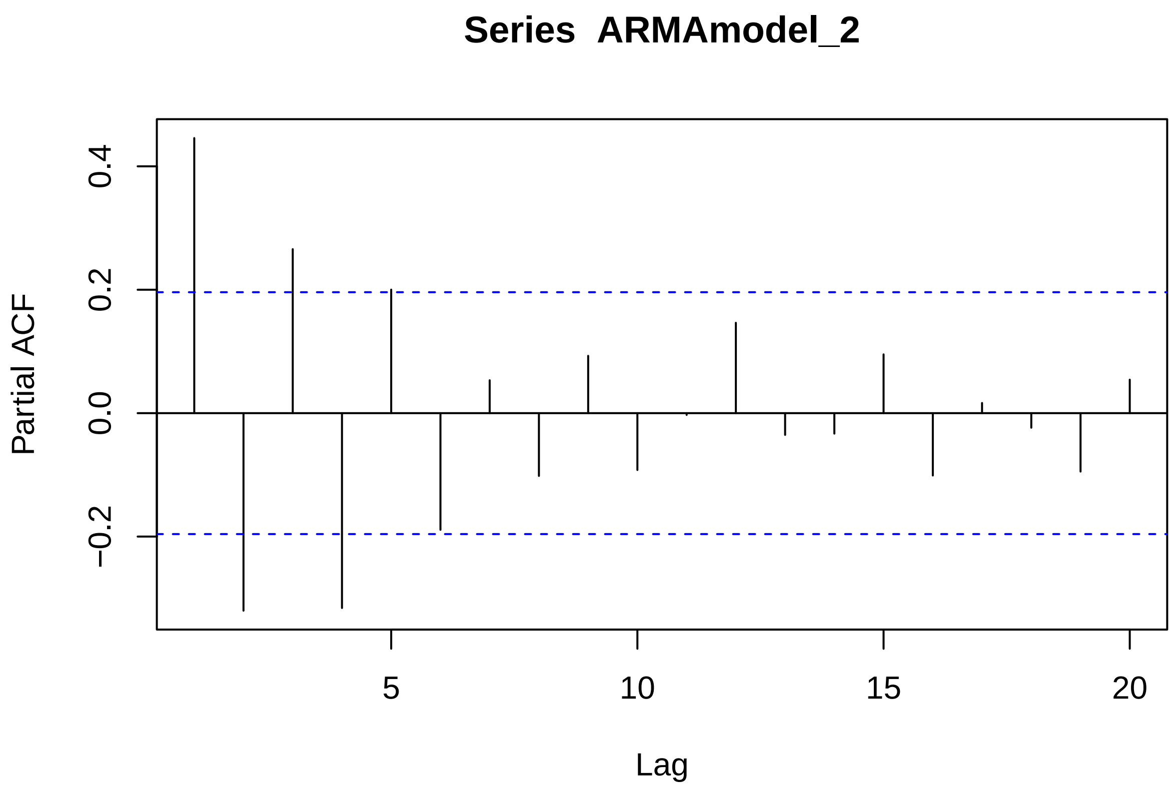

```
print(ARMA10_pacf$acf)
```

```
## , , 1
##
##              [,1]
##  [1,]  0.671923984
##  [2,] -0.087276531
##  [3,]  0.238525254
##  [4,] -0.006633578
##  [5,] -0.020809410
##  [6,] -0.067059216
##  [7,]  0.105979621
##  [8,] -0.010315911
##  [9,] -0.182011247
## [10,]  0.118394026
## [11,]  0.041013345
## [12,]  0.028123791
## [13,] -0.023405017
## [14,] -0.039063475
## [15,] -0.164769072
## [16,] -0.043191203
## [17,] -0.034308398
## [18,] -0.051619522
## [19,] -0.018760269
## [20,]  0.107659643
```

```r
ARMA10 <- arima(ARMAmodel_1, order = c(1,0,0))
coef_ARMA10 <- ARMA10$coef; coef_ARMA10
```

```
##       ar1 intercept
## 0.6714151 0.1814781
```

```r
#ARMA(0,1)
ARMA01_pacf <- pacf(ARMAmodel_2)
```




```r
print(ARMA01_pacf$acf)
```

```
## , , 1
##
##               [,1]
##  [1,]  0.445762216
##  [2,] -0.320083053
##  [3,]  0.265773537
##  [4,] -0.315776667
##  [5,]  0.200172430
##  [6,] -0.189078665
##  [7,]  0.053310765
##  [8,] -0.101765858
##  [9,]  0.092844399
## [10,] -0.092114940
## [11,] -0.002885224
## [12,]  0.146386544
## [13,] -0.035218367
```

```
## [14,] -0.033158134
## [15,]  0.095215864
## [16,] -0.101067089
## [17,]  0.016517501
## [18,] -0.023607553
## [19,] -0.094677827
## [20,]  0.054290185
```

```
## can not use the PACF
ARMA01 <- arima(ARMAmodel_2, order = c(0,0,1))
coef_ARMA01 <- ARMA01$coef; coef_ARMA01
```

```
##        ma1  intercept
## 0.90150135 0.03247681
```

```
#ARMA(1,1)
ARMA11 <- arima(ARMAmodel_3, order = c(1,0,1))
coef_ARMA11 <- ARMA11$coef; coef_ARMA11
```

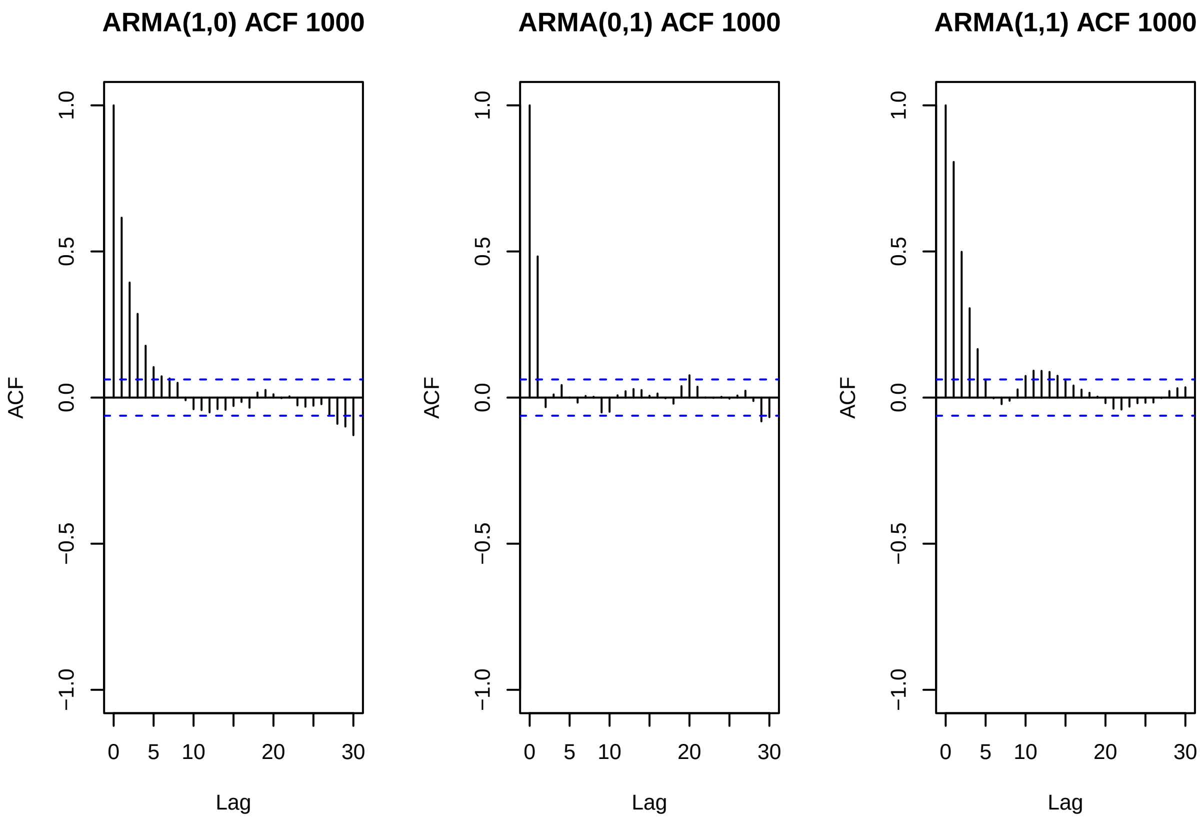
```
##        ar1        ma1  intercept
##  0.6072317  0.9266740 -0.1684434
```

Answer: They do not 100% same, but close. For ARMA(1,0), the computed value (0.67) is close to the coefficient (0.6). For ARMA(0,1), the computed value (0.9015) is close to the coefficient (0.9). For ARMA(1,1), the computed AR value is 0.607 and MA value is 0.927.

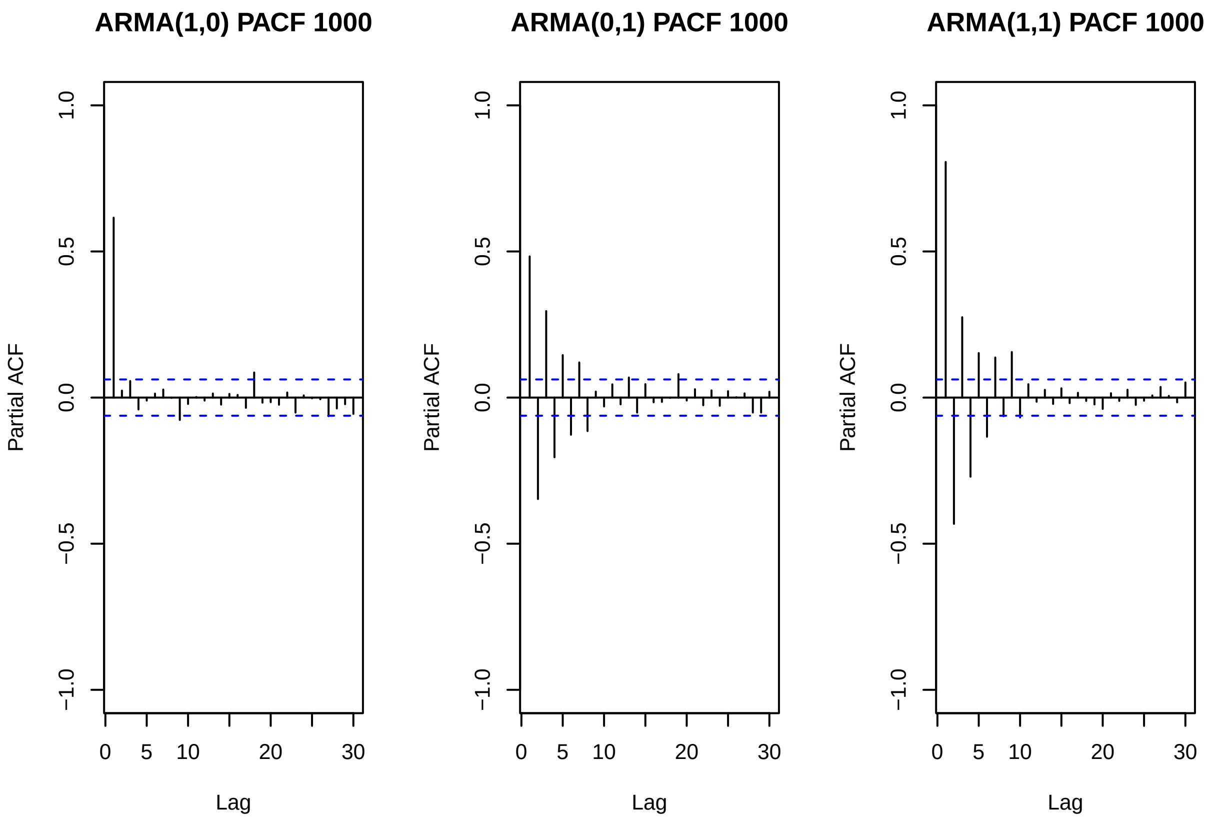
(e) Increase number of observations to $n = 1000$ and repeat parts (a)-(d).

```
#ARMA(1,0)
set.seed(111)
ARMAmodel_1_1000<- arima.sim(model=list(ar=0.6), n=1000) #the AR coefficient is 0.6
#ARMA(0,1)
set.seed(222)
ARMAmodel_2_1000<- arima.sim(model=list(ma=0.9), n=1000) #the MA coefficient is 0.9
#ARMA(1,1)
set.seed(333)
ARMAmodel_3_1000<- arima.sim(model=list(ar=0.6,ma=0.9), n=1000)

par(mfrow = c(1,3))
acf(ARMAmodel_1_1000, main = "ARMA(1,0) ACF 1000", ylim=c(-1,1))
acf(ARMAmodel_2_1000, main = "ARMA(0,1) ACF 1000", ylim=c(-1,1))
acf(ARMAmodel_3_1000, main = "ARMA(1,1) ACF 1000", ylim=c(-1,1))
```




```r
par(mfrow = c(1,3))
pacf(ARMAmodel_1_1000, main = "ARMA(1,0) PACF 1000", ylim=c(-1,1))
pacf(ARMAmodel_2_1000, main = "ARMA(0,1) PACF 1000", ylim=c(-1,1))
pacf(ARMAmodel_3_1000, main = "ARMA(1,1) PACF 1000", ylim=c(-1,1))
```




```r
#ARMA(1,0)
ARMA10_1000_pacf <- pacf(ARMAmodel_1_1000)
print(ARMA10_1000_pacf$acf)
```

```
## , , 1
##
##              [,1]
##  [1,]  0.615496530
##  [2,]  0.023888859
##  [3,]  0.056761300
##  [4,] -0.040718820
##  [5,] -0.010210595
##  [6,]  0.013761635
##  [7,]  0.027542204
##  [8,] -0.001165630
##  [9,] -0.076545091
## [10,] -0.021874890
## [11,]  0.002008580
## [12,] -0.010304526
## [13,]  0.013987126
## [14,] -0.024117615
## [15,]  0.012449658
## [16,]  0.009001410
## [17,] -0.035071775
## [18,]  0.085859244
## [19,] -0.017345874
## [20,] -0.015830512
```

```
## [21,] -0.024681976
## [22,]  0.017460590
## [23,] -0.051335161
## [24,]  0.007434116
## [25,] -0.002511619
## [26,] -0.005875369
## [27,] -0.063777768
## [28,] -0.037385853
## [29,] -0.022773226
## [30,] -0.055979126
```

```
ARMA10_1000 <- arima(ARMAmodel_1_1000, order = c(1,0,0))
coef_ARMA10_1000 <- ARMA10_1000$coef; coef_ARMA10_1000
```

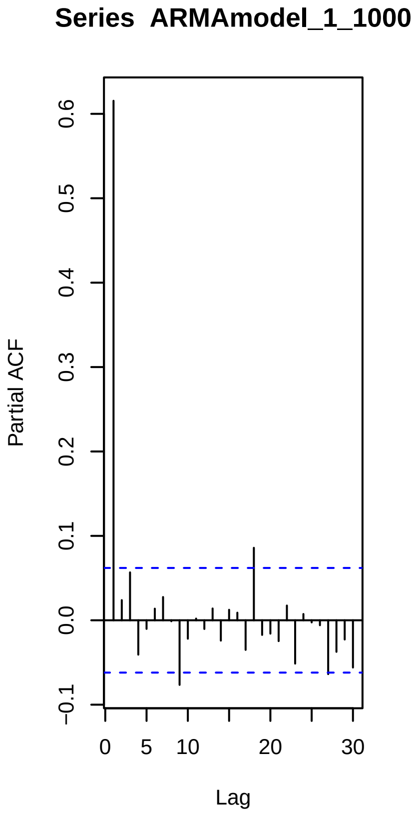
```
##        ar1  intercept
## 0.61541644 0.03160003
```

```
#ARMA(0,1)
ARMA01_1000 <- arima(ARMAmodel_2_1000, order = c(0,0,1))
coef_ARMA01_1000 <- ARMA01_1000$coef; coef_ARMA01_1000
```

```
##        ma1  intercept
##  0.8946308 -0.0297921
```

```
#ARMA(1,1)
ARMA11_1000 <- arima(ARMAmodel_3_1000, order = c(1,0,1))
coef_ARMA11_1000 <- ARMA11_1000$coef; coef_ARMA11_1000
```

```
##        ar1        ma1  intercept
## 0.61952188 0.91071842 0.08699625
```



Answer: Compared with before (n=100), the blue line area (non-significant interval) is now smaller, the difference between the significant value and the subsequent non-significant value is more obvious, and it is easier to determine the cut-off. Therefore, based on these plots, it can correctly identify the model type and order. They do not 100% same, but close. For ARMA(1,0)_1000, the computed value (0.620) is more closer to the coefficient (0.6). For ARMA(0,1), the computed value is 0.895. For ARMA(1,1), the computed AR value is 0.620 and MA value is 0.911.

Q3

Consider the ARIMA model $y_t = 0.7 * y_{t-1} - 0.25 * y_{t-12} + a_t - 0.1 * a_{t-1}$

(a) Identify the model using the notation ARIMA$(p, d, q)(P, D, Q)_s$, i.e., identify the integers p, d, q, P, D, Q, s (if possible) from the equation.

Answer: Due to $0.7 * y_{t-1}$, p = 1 Due to$-0.25 * y_{t-12}$, P=1 Due to$-0.1 * a_{t-1}$ q = 1, Q = 0 Due to no $y_t-y_{t-s} and y_t-Y_{t-1}, d= 0, D=0. s= 12. p,d,q,P,D,Q,s 1,0,1,1,0,0,12

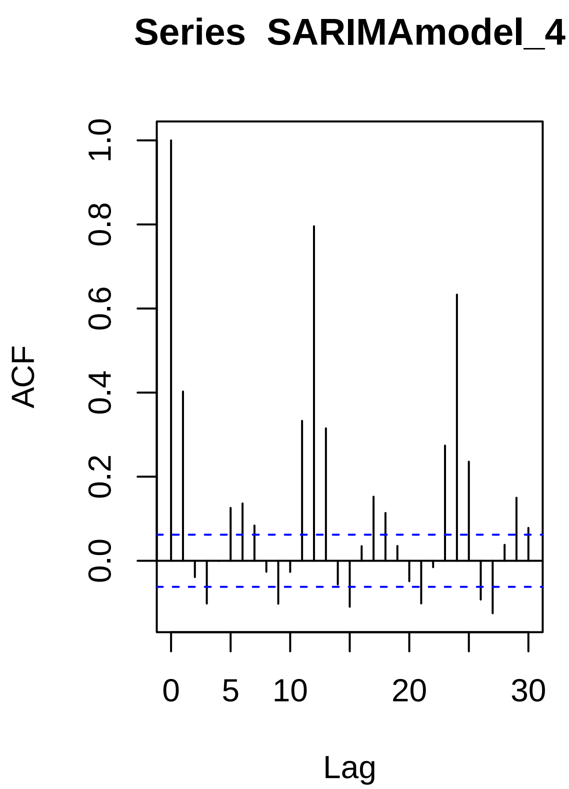
(b) Also from the equation what are the values of the parameters, i.e., model coefficients.

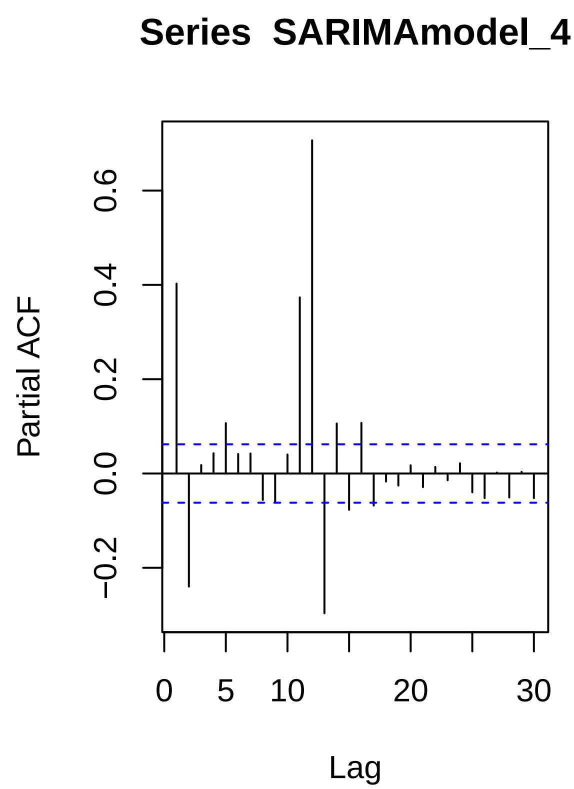
Answer: phi = 0.7; AR coefficients. theta = 0.1; MA coefficients. PHI = -0.25; SAR coefficients.

Q4

Plot the ACF and PACF of a seasonal ARIMA$(0, 1) \times (1, 0)_{12}$ model with $\phi = 0.8$ and $\theta = 0.5$ using R. The 12 after the bracket tells you that $s = 12$, i.e., the seasonal lag is 12, suggesting monthly data whose behavior is repeated every 12 months. You can generate as many observations as you like. Note the Integrated part was omitted. It means the series do not need differencing, therefore $d = D = 0$. Plot ACF and PACF for the simulated data. Comment if the plots are well representing the model you simulated, i.e., would you be able to identify the order of both non-seasonal and seasonal components from the plots? Explain.

```
SARIMAmodel_4<- sim_sarima(model=list(ma=0.5,sar=0.8, nseasons=12), n=1000)
par(mfrow = c(1, 2))
acf(SARIMAmodel_4)
pacf(SARIMAmodel_4)
```





>

Answer: The ACF plot shows a significant spike at lag 12 and lag 24, which suggests the presence of a seasonal component. Meanwhile, the cut-off after lag 1 can help to identify the order of non-seasonal MA as 1. In PACF, the order of seasonal AR might be able to identify due to lag 12 have high and significant value or there is single spike in lag 12.